In re Patent Application of:
WESTPHAL
Serial No. **09/787,290**
Filed: **JUNE 28, 2001**
_____/

In the Claims:

Please amend the above-identified application as follows:

Claim 1 (original) A method of designing logical circuits, comprising the steps of:

a. representing the logic of a logical circuit to be designed as points and vectors in a vector space; and

b. using the points and vectors in a vector space to simplify the logic of the logical circuit to a simpler form; and

c. designing the logical circuit using the simpler form.

Claim 2 (original) A method of manufacturing logical circuits, comprising the steps of:

a. representing the logic of a logical circuit to be manufactured as points and vectors in a vector space; and

b. using the points and vectors in a vector space to simplify the logic of the logical circuit to a simpler form; and

c. using the simpler form to implement the logical circuit in hardware.

Claim 3 (original) A method of simplifying logical circuits, comprising the steps of:

a. representing the logic of a logical circuit as points and vectors in a vector space; and

b. modifying the representation in vector space using at least one process rule of a set of process rules to simplify the logic.

Claim 4 (currently amended) The method of claim 3 in which at least one process rule of a set of process rules consists of one of the following process rules:

a. Process Rule 1--

a1. Represent the alternational normal schema, the target schema **t**, as a set of vectors in the ANS-space,

a2. Each clause or disjunct of **t** is a position vector (i.e. one pointing to **O**) with **O** at one corner of a set of parallelograms made of propositional addresses to the ι-point at the other,

a3. Any two other outside vertices of such a parallelogram are implicants which are among the original clauses of **t**;

b. Process Rule 2—

b1. Pick any two clauses,

b2. If there is a propositional address σ at the midpoint between the component clauses, the vector from ι to σ, is the simplification of and can replace the relevant clauses of **t**;

c. Process Rule 3—

c1. Generate ι implicants until each clause or vector has been used at least once,

c2. If a disjunct **d** of **t** cannot be used because it forms no propositional address with any other disjunct, then **d** must appear unmodified in the final schema which is the simplification of **t**;

d. Process Rule 4—

d1. If an ι point exists in **t**, delete the vectors which produce it in favor of the vector from ι to **O**;

e. Process Rule 5—

e1. For a clause in a schema which subsumes another clause eliminate the subsuming clause;

f. Process Rule 6—

f1. Couples such as $\mathbf{pq} \vee \bar{p}\,\bar{q}$ or $\overline{pq}\,\mathbf{s} \vee \mathbf{pq}\bar{s}$ cannot be summed to zero at the origin;

g. Process Rule 7—

g1. Translate vectors if a corresponding σ-point exist for a ι-point then σ is the simplification of ι;

g2. Any superpositions of parallel arrows in opposite directions represent equivalences,

g3. For equivalences, (a) drop the longer clause at either end of any doubleheaded arrow, (b) drop pairs, triples etc. of double-headed arrows which meet at a point in favor of the vector from that point to **O** and (c) drop a vector or clause in the target schema which is itself the resultant of any other two vectors;

h. Process rule 8—

h1. A simplification is complete if in the system which replaces the target schema no vectors or clauses are subsumed by others and no double-headed vectors remain.

Claim 5 (original) Apparatus for simplifying logical circuits, comprising:

a. a processing element configured to represent the logic of a logical circuit to be simplified as points and vectors in a vector space and to use the points and vectors to simplify the logic of the logical circuit to a simpler form.

Claim 6 (original) The apparatus of claim 5 in which the processing element is an optical computer.

Claim 7 (original) The apparatus of claim 5 in which the processing element is a digital computer.

Claim 8 (original) The apparatus of claim 1 in which the processing element is an colorimetric computer.

Claim 9 (original) The apparatus of claim 1 in which the processing element is an analog computer.

Claim 10 (original) A computer program product, comprising:

a. a memory element; and

b. a computer program stored on said memory medium, said computer program comprising instructions for representing the logic of a logical circuit to be designed as points and vectors in a vector space and for using the points and vectors in a vector space to simplify the logic of the logical circuit to a simpler_ form and for designing the logical circuit using the simpler form.

Claim 11 (original) A computer program product, comprising:

a. a memory element; and

b. a computer program stored on said memory medium, said computer program comprising instructions for representing the logic of a logical circuit to be manufactured as points and vectors in a vector space, and for using the points and vectors in a vector space to simplify the logic of the logical circuit to a simpler form, and for using the simpler form to implement the logical circuit in hardware.

Claim 12 (original) A computer program product, comprising:

a. a memory element; and

b. a computer program stored on said memory medium, said computer program comprising instructions for representing the logic of a logical circuit as points and vectors in a vector space, and for modifying the representation in a vector space using at least one process rule of a set of process rules to simplify the logic.